

Draft

ANALYSIS OF LDAP v3 STANDARDS

LAST UPDATED: 25 Feb 2000

Draft

Draft

Contents

1	Introduction	1
2	Scope	1
3	Instructions.....	1
3.1	Prerequisite notation	1
3.2	Status column	1
3.3	Support column.....	2
3.4	Predicate Name	2
3.5	Note column.....	2
4	Protocol Parsing	3
4.1	General Capabilities	3
4.2	LDAPMessage	4
Ref.	RFC 2251 Para 4.1.1	4
4.3	Operations	4
4.4	Unsolicited Notification	4
4.5	Bind.....	5
4.6	Unbind.....	5
4.7	Search.....	6
4.7.1	Filter	7
4.7.1.1	SubStringFilter	7
4.7.1.2	MatchingRuleAssertion.....	7
4.7.1.3	AttributeValueAssertion	7
4.8	Modify.....	8
4.9	Add.....	8
4.10	Delete.....	9
4.11	Modify DN Operation.....	9
4.12	Compare Operation.....	9
4.13	Abandon Operation	10
4.14	Extended Operation	10
4.15	LDAP Result	11
4.16	Unsolicited Notification	12
4.16.1	Notice of Disconnection	12
4.17	Syntax	13
4.17.1	Syntax Description.....	14
4.17.2	RFC 1274Syntax	14
4.18	Attributes Types	15
4.18.1	Operational Attribute	15
4.18.1.1	Subschema Entry	15
4.18.1.1.1	Matching Rule Description	15
4.18.1.1.2	Matching Rule Use Description.....	16
4.18.1.1.3	DIT Structure Rules Description	16
4.18.1.1.4	Name Form Description.....	16
4.18.1.1.5	DIT Content Rule Description	17
4.18.2	Attribute Type Description	17
4.18.2.1	Syntax Object Identifier.....	17
4.18.2.2	Attribute Usage	18
4.18.3	User Attribute	19
4.18.4	RFC 1274 Attribute	21

Draft

4.19	Object Classes.....	22
4.19.1	Object Classes Description	23
4.19.2	PKI Object Classes.....	23
4.19.3	RFC 1274 Object Classes.....	24
4.19.4	Matching Rules.....	25
4.20	Rules for DN Encoding	26
4.21	LDAP URL Definition.....	27
4.21.1	LDAP URL Extension.....	27

1 Introduction

This document, when completed will parse all of the LDAP version 3 related documents into their protocol and data elements. This document can be used to test commercial products against features defined in the standard.

This document is a work in progress. When completed it will contain the elements associated with the LDAP protocol.

The following is a list of LDAP related documents which have been parsed thus far:

- RFC 1274
- RFC 2251
- RFC 2252
- RFC 2253
- RFC 2254
- RFC 2255
- RFC 2256
- [RFC 2587](#)

2 Scope

This document only covers the LDAP version 3 protocol. While some version 3 implementations may be capable of also supporting LDAP version 2, such a parsing is outside of the scope of this document.

3 Instructions

3.1 Prerequisite notation

If a predicate applies to a whole table, a prerequisite line may be specified in front of the table to which it applies. A prerequisite line takes the form:

Prerequisite: <Predicate>

The meaning of such a line is that if <predicate> is True, then the table applies, else it is not applicable.

3.2 Status column

This column indicates the level of support required.

The values are as follows:

- m the capability is required to be implemented, in conformance with the related specification;
- o the capability may be implemented, and if it is implemented it is required to conform to the related specification
- c the requirement on the capability depends on the selection of other optional or conditional items;
- i the capability is outside the scope of this document, and hence irrelevant and not subject to conformance testing;
- in the given context it is impossible to use this capability.

Nested conditionals are denoted by nested numbering (e.g. 1, 1.1, 1.1.1, etc.) of the item descriptions in the tables. A table may have zero, one or more levels of nesting. The status of a leading item is specified by its status entry, as defined above. The status of a subordinate (that is nested) item is specified as follows: If the superior item is supported, the status of the subordinate item is determined by its status column entry

Draft

and applicable predicate, if any. If the superior item is not supported, the subordinate item is not applicable, independent of its status column entry.

3.3 Support column

An item is not considered implemented simply because a default value has been defined by the standard. In order for an Implementation Under Test (IUT) to claim a protocol element is implemented, it must have the ability, where appropriate, to generate, receive, and perform the appropriate action,

The support values are:

- Y yes, the item has been implemented;
- N no, the item has not been implemented;
- the item is not applicable;

3.4 Predicate Name

The predicate name indicates that name upon which the predicate is based. A predicate name flagged with an asterisk preceding the predicate name indicates the condition by which the predicate is being set. A predicate name not flagged with an asterisk indicates the predicate on which the conditional support is based.

3.5 Note column

This column indicates the following:

- notxx refers to Note xx
- d(xx) a default value xx within () is defined in the Standard. When absent in the PDU, both sender and receiver shall interpret it as having the default value specified in the standard
- See xx refers to Table xx
- = xx indicates the value that must be contained in a protocol element

4 Protocol Parsing

4.1 General Capabilities

Item No.	Protocol Element	Server		Predicate Name	Note
		Status	Support		
1	LDAP over TCP	m			
1.1	Port 389	o.1			
1.2	Another Port	o.1			
2	BER	m			
2.1	Only definite form of length encoding	m			
2.2	Only primitive form encoding for OCTET STRINGS	m			
2.3	BOOLEAN TRUE type encoded only as 0xFF	m			
2.4	default values must be absent	m			
3	Disconnect for improper LDAPMessage (RFC 2251 4.1.1)	m			
4	Return result code of protocolError for non-parsed requests (RFC 2251 4.1.1)	m			
5	shadowing servers	o			
5.1	Does not violate access control constraints defined in master	c:m			
6	Ignores elements of sequence encoding with unrecognizable tags	m			
7	Implement additional attribute types	o			
8	Attribute types follow description in 4.4.4 of RFC 2251	m			
9	Publish attributeTypes in the subschemasubentry attributeTypes attribute (RFC 2252 4.2)	o			
10	Implement additional object classes (RFC 2252 4.4)	o			
11	Publish object classes in the subschemasubentry object class attribute	o			
12	Implement additional matching rules (RFC 2252 4.5)	o			
13	Publish matchingrules in the subschemasubentry matchingrules attribute	o			
14	Allow clients to modify subschema entries	o		*SubschemaMod	
15	Support for \ escaping mechanism (RFC 2252 4.3)	o			
16	Follow rules defined for encoding of Distinguished Names (RFC 2253)	m			See 4.20
17	Support for LDAP URL format and resolution (RFC 2255)	m			See 4.21
18	Support for attribute subtyping	o		*att_subtype	
19	Support for alias	o		*alias	

o.1: Either the well known port (389) and/or another port must be implemented.

Draft

4.2 LDAPMessage

Ref. RFC 2251 Para 4.1.1

Item No.	Protocol Element	Server		Predicate Name	Note
		Status	Support		
1	messageID	m			Note 1
2	protocolOp	m			See 4.3
3	controls	o			
4	controlType	m			Note 2
5	criticality	m			Note 3,4
6	controlValue	m			

Note 1: If improper messageID then server must return notice of disconnect with result code of protocolError.

Note 2: Must UTF8 encoded dotted decimal representation of control OID.

Note 3: If TRUE and control type is unrecognized or inappropriate, server must not perform operation and return resultCode unsupportedCriticalExtension.

Note 4: If FALSE and control type is unrecognized or inappropriate server must ignore the control.

4.3 Operations

Ref. RFC 2251 Para 4.1

Item No.	Protocol Element	Server		Predicate Name	Note
		Status	Support		
1	Bind	m			
2	Unbind	m			
4	Search	m			
5	modify	m			
6	add	m			
7	delete	m			
8	modifyDN	m			
9	compare	m			
10	abandon	m			
11	extended	m			Note 1

Note 1: Extended operation protocol elements must be supported but the server need not support any specific extended operations.

4.4 Unsolicited Notification

Ref. RFC 2251 Para 4.1

Item No.	Protocol Element	Server		Predicate Name	Note
		Status	Support		
1	Notice of disconnection	o		*NoticeDisconnect	See 4.16.1

Draft

4.5 Bind

Servers must treat operations prior to a bind as unauthenticated.

Ref. RFC 2251 Para 4.2

Item No.	Protocol Element	SERVER		Predicate	Note
		Status	Support		
1.	BindRequest	m			Note 1
1.1.	APPLICATION	m			= 0
1.2.	version	m			Note 2
1.3.	name	m			Note 3
1.4.	authentication	m			
1.4.1.	simple	m			Note 4
1.4.2.	sasl	m			Note 5
1.4.2.1.	mechanism	m			Note 5
1.4.2.2.	credentials	m			Note 5
2.	BindResponse	m			
2.1.	APPLICATION	m			= 1
2.2.	LDAPResult	m			See 4.15
2.3.	serverSaslCreds	o			Note 6

Note 1: Failed binds will be treated as anonymous.

Note 2: If version is 2 server may support LDAP v2 model. If NULL then default version is LDAP v3.

Note 3: May be NULL for anonymous binds, authentication performed at lower layer or SASL credentials that include the LDAPDN

Note 4: NULL for not anonymous binds otherwise contains a cleartext password.

Note 5: Used if server authentication or challenge-response authentication.

Note 6: Server must support the sasl protocol elements however, they are not obligated to support all sasl mechanisms.

4.6 Unbind

Ref. RFC 2251 Para 4.3

Item No.	Protocol Element	SERVER		Predicate	Note
		Status	Support		
1	UnbindRequest	m			Note 1
1.1	APPLICATION	m			= 2

Note 1: Server may assume terminated session, discard outstanding requests, and close connection.

Draft

4.7 Search

Ref. RFC 2251 Para 4.5

Item No.	Protocol Element	SERVER		Predicate	Note
		Status	Support		
1.	SearchRequest	m			
1.1.	APPLICATION	m			=3
1.2.	baseObject	m			
1.3.	scope	m			
1.3.1.	baseObject	m			
1.3.2.	singleLevel	m			
1.3.3.	wholeSubtree	m			
1.4.	derefAliases	m			
1.4.1.	neverDereAliases	m			
1.4.2.	derefInSearching	m			
1.4.3.	derefFindingBaseObj	m			
1.4.4.	derefAlways	m			
1.5.	sizeLimit	m			Note 1
1.6.	timeLimit	m			Note 2
1.7.	typesOnly	m			Note 3
1.8.	filter	m			See 4.7.1
1.9.	attributes	m			
2.	SearchResponse	m			
2.1.	SearchResultEntry	m			Note 4
2.1.1.	Application	m			=4
2.1.2.	objectName	m			
2.1.3.	attributes	m			
2.1.4.	type	m			
2.1.5.	vals	m			
2.2.	SearchResultReference	m			Note 4, 5
2.2.1.	APPLICATION	m			=19
2.2.2.	LDAPURL	m			Note 6
2.3.	SearchResultDone	m			
2.3.1.	APPLICATION	m			=5
2.3.2.	LDAPResult	m			See 4.15

Note 1: Zero size limit indicates no size restrictions.

Note 2: Zero time limit indicates no time restrictions.

Note 3: TRUE returns attribute types only. FALSE returns attributes types and values.

Note 4: Result may return zero or more responses for each search request. (RFC 2251 4.5.2)

Note 5: Not returned if baseObject was not located. Referral returned instead.

Note 6: The DN must be present in the URL with the new target object name.

Draft

4.7.1 Filter

Implementations claiming conformance for search filter must support string representation as defined in RFC 2254 paragraph 4.

Ref. RFC 2251 Para 4.5.1

Item No.	Protocol Element	SERVER		Predicate	Note
		Status	Support		
1	and	m			
2	or	m			=3
3	not	m			
4	equalityMatch	m			See 4.7.1.3, Note 1
5	substrings	m			See 4.7.1.1
6	greaterOrEqual	m			See 4.7.1.3, Note 2
7	lessOrEqual	m			See 4.7.1.3, Note 3
8	present	m			
9	approxMatch	m			See 4.7.1.3, Note 4
10	extensibleMatch	m			See 4.7.1.2

Note 1: Equivalent to ABNF for simple item with equal filetype defined in RFC 2254 paragraph 4.

Note 2: Equivalent to ABNF for simple item with greater filetype defined in RFC 2254 paragraph 4.

Note 3: Equivalent to ABNF for simple item with less filetype defined in RFC 2254 paragraph 4.

Note 4: Equivalent to ABNF for simple item with approx filetype defined in RFC 2254 paragraph 4.

4.7.1.1 SubStringFilter

Ref. RFC 2251 Para 4.5.1

Item No.	Protocol Element	SERVER		Predicate	Note
		Status	Support		
1	type	m			
2	substrings	m			
2.1	initial	m			
2.2	any	m			
2.3	final	m			

4.7.1.2 MatchingRuleAssertion

Ref. RFC 2251 Para 4.5.1

Item No.	Protocol Element	SERVER		Predicate	Note
		Status	Support		
1	matchingRule	m			
2	type	m			= 3
3	matchValue	m			
4	dnAttributes	m			default false

4.7.1.3 AttributeValueAssertion

Ref. RFC 2251 Para 4.1.7

Item No.	Protocol Element	SERVER		Predicate	Note
		Status	Support		
1.	attributeDesc	m			
2.	attributeValue	m			Note 1

Note 1: Values containing characters; * () \ and NUL should conform to RFC 2254 paragraph 4 which describes the required escape sequences.

Draft

4.8 Modify

Ref. RFC 2251 Para 4.6

Item No.	Protocol Element	Server		Predicate	Note
		Status	Support		
1	ModifyRequest	m			
1.1	APPLICATION	m			= 6
1.2	object	m			Note 1
1.3	modification	m			Note 2, 3
1.3.1	operation	m			
1.3.1.1	add	m			
1.3.1.2	delete	m			Note 4
1.3.1.3	replace	m			Note 5
1.3.2	modification	m			
1.3.2.1	type	m			Note 6
1.3.2.2	vals	m			
2	ModifyResponse	m			
2.1	APPLICATION	m			= 7
2.2	LDAPResult	m			See 4.15

Note 1: Contains the DN of the object to be modified.

Note 2: Contains the list of modification to be performed. List must be done in order.

Note 3: May not be used to replace any value associated with a DN.

Note 4: Remove attribute if no values are listed or list contains all values of the attribute.

Note 5: If attribute does not exist then create it. If no value is given the attribute is deleted.

Note 6: If attribute does not have an equality match filter associated with it then values can not be removed using the delete form of modification but the replace form must be used instead.

4.9 Add

Ref. RFC 2251 Para 4.7

Item No.	Protocol Element	SERVER		Predicate	Note
		Status	Support		
1	AddRequest	m			
1.1	APPLICATION	m			= 8
1.2	entry	m			Note 1, 2
1.3	attributes	m			Note 3
1.3.1	type	m			
1.3.2	vals	m			Note 4
2	AddResponse	m			
2.1	APPLICATION	m			= 9
2.2	LDAPResult	m			See 4.15

Note 1: The DN of the entry being added. Aliases are not dereferenced during this operation.

Note 2: The parent entry must already exist to create the entry. If the parent exists on another server then a referral should be returned.

Note 3: List of attributes which make up the entry.

Note 4: Value for this entry must not already exist.

Draft

4.10 Delete

Ref. RFC 2251 Para 4.8

Item No.	Protocol Element	Server		Predicate	Note
		Status	Support		
1	DelRequest	m			
1.1	APPLICATION	m			= 10
1.2	LDAPDN	m			Note 1
2	DelResponse	m			
2.1	APPLICATION	m			= 11
2.2	LDAPResult	m			See 4.15

Note 1: Consists of DN, server will not dereference aliases while resolving name of target entry to be removed, leaf entries without subordinates can be deleted.

4.11 Modify DN Operation

Ref. RFC 2251 Para 4.9

Item No.	Protocol Element	SERVER		Predicate	Note
		Status	Support		
1	ModifyDNRequest	m			
1.1	APPLICATION	m			= 12
1.2	entry	m			Note 1
1.3	newrdn	m			Note 2
1.4	deleteoldrdn	m			Note 3
1.5	newSuperior	m			Note 4
2	ModifyDNResponse	m			
2.1	APPLICATION	m			= 13
2.2	LDAPResult	m			See 4.15

Note 1: DN entry to be changed may or may not have subordinates.

Note 2: RDN will form the leftmost component of the new entry.

Note 3: If TRUE deleteoldrdn attributes, if FALSE retain as attributes of the entry

Note 4: Movement of entries and subtrees between servers need not be supported.

4.12 Compare Operation

Ref. RFC 2251 Para 4.10

Item No.	Protocol Element	SERVER		Predicate	Note
		Status	Support		
1	CompareRequest	m			
1.1	APPLICATION	m			= 14
1.2	entry	m			Note 1
1.3	ava	m			
2	CompareResponse	m			
2.1	APPLICATION	m			= 15
2.2	LDAPResult	m			See 4.15

Note 1: Contains the DN of the entry.

Draft

4.13 Abandon Operation

Ref. RFC 2251 Para 4.11

Item No.	Protocol Element	SERVER		Predicate	Note
		Status	Support		
1	AbandonRequest	m			
1.1	APPLICATION	m			=16
1.2	MessageID	m			Note 1

Note 1: Servers must discard abandon requests for messageID not recognized in operations that have been abandoned.

4.14 Extended Operation

Ref. RFC 2251 Para 4.12

Item No.	Protocol Element	SERVER		Predicate	Note
		Status	Support		
1	ExtendedRequest	m			
1.1	APPLICATION	m			=23
1.2	requestName	m			Note 1
1.3	requestValue	m			
2	ExtendedResponse	m			
2.1	APPLICATION	m			=24
2.2	LDAPResult	m			See 4.15
2.3	responseName	c1			
2.4	response	c1			

Note 1: The server does not recognize the extended request then return LDAP result of protocol error.

c1: If the server supports an extended operation for which a response has been defined then support for this item is m else o.

Draft

4.15 LDAP Result

Ref. RFC 2251 Para 4.1.10

Item No.	Protocol Element	SERVER		Predicate	Note
		Status	Support		
1	resultCode	m			
1.1	success	m			Note 1
1.2	operationsError	m			Note 2
1.3	protocolError	m			Note 2,9,10
1.4	timeLimitExceeded	m			
1.5	sizeLimitExceeded	m			
1.6	compareFalse	m			Note 8
1.7	compareTrue	m			Note 8
1.8	authMethodNotSupported	m			Note 2
1.9	strongAuthRequired	m			Note 2,10
1.10	referral	m			Note 2, 3, 5
1.11	adminLimitExceeded	m			
1.12	unavailableCriticalExtension	m			
1.13	confidentialityRequired	m			
1.14	saslBindInProgress	m			Note 2
1.15	nosuchAttribute	m			Note 8
1.16	undefinedAttributeType	m			
1.17	inappropriateMatching	m			
1.18	constraintViolation	m			
1.19	attributeOrValueExists	m			
1.20	invalidAttributeSyntax	m			
1.21	noSuchObject	m			Note 5
1.22	aliasProblem	m			
1.23	invalidDNSyntax	m			
1.24	aliasDereferencingProblem	m			
1.25	inappropriateAuthentication	m			Note 2
1.26	invalidCredentials	m			Note 2
1.27	insufficientAccessRights	m			
1.28	busy	m			
1.29	unavailable	m			Note 2,10
1.30	unwillingToPerform	m			
1.31	loopDetect	m			
1.32	namingViolation	m			
1.33	objectClassViolation	m			
1.34	notAllowedOnNonLeaf	m			
1.35	notAllowedOnRDN	m			Note 4
1.36	entryAlreadyExists	m			Note 7
1.37	objectClassModsProhibited	m			
1.38	affectsMultipleDSAs	m			Note 7
1.39	other	m			
2	matchedDN	m			Note 5
3	errorMessage	m			
4	referral	m			Note 2, 3, 5, 11
4.1	LDAPURL	m			

Note 1: Used in all operation responses.

Note 2: Used in Bind Response (RFC 2251 4.2.3).

Note 3: Used in SearchResultDone (RFC 2251 4.5.2)

Note 4: Used in Modify response (RFC 2251 4.6)

Draft

- Note 5: Used in Add response (RFC 2251 4.7)
Note 6: Used in Delete response (RFC 2251 4.8)
Note 7: Used in ModifyDN response (RFC 2251 4.9)
Note 8: Used in CompareResponse (RFC 2251 4.10)
Note 9: Used in ExtendedResponse (RFC 2251 4.12)
Note 10: Used in Notice of Disconnection Response(RFC 2251 4.4.1)
Note 11: The referral field is only present when the result code field value is referral (item 1.10).

4.16 Unsolicited Notification

Unsolicited Notification is a LDAP Message in the form of an ExtendedResponse with MessageID equal to zero. The responseName field must contain the OID of the Unsolicited Notification.

4.16.1 Notice of Disconnection

After sending this notice the server must close the connection.

Prerequisite: [NoticeDisconnect]

Ref. RFC 2251 Para 4.4.1

Item No.	Protocol Element	SERVER		Predicate	Note
		Status	Support		
1	ExtendedResponse	m			
1.1	APPLICATION	m			=24
1.2	LDAPResult	m			See 4.15
1.3	responseName	m			Note 1
1.4	response	-			Note 2

Note 1: LDAPOID equal to 1.3.6.1.4.1.1466.20036

Note 2: Response field is absent.

Draft

4.17 Syntax

Ref. RFC 2252 Para 4.3.2

Item No.	Protocol Element	Server		Predicate Name	Note
		Status	Support		
1.	ACI Item	o			Note 1
2.	Access Point	o			
3.	AttributeType Description	m			See 4.18.2
4.	Audio	o			Note 1
5.	Binary	m			Note 1, 2
6.	Bit String	o			Note 3
7.	Boolean	o			Note 3
8.	Certificate	o			Note 1, 3
9.	Certificate List	o			Note 1, 3
10.	Certificate Pair	o			Note 1, 3
11.	Country String	o			Note 3
12.	DN	m			
13.	Data Quality Syntax	o			
14.	Delivery Method	o			
15.	Directory String	o			Note 3, 4
16.	DIT Content Rule Description	c1			Note 3
17.	DIT Structure Rule Description	c2			
18.	DLSumit Permission	o			
19.	DSA Quality Syntax	o			
20.	DSE Type	o			
21.	Enhanced Guide	o			
22.	Facsimile Telephone Number	o			Note 3
23.	Fax	o			Note 3
24.	Generalized Time	m			
25.	Guide	o			
26.	IA5 String	c4			Note 3
27.	INTEGER	m			
28.	JPEG	o			Note 3
29.	LDAP Syntax Description	o			Note 3
30.	LDAP Schema Definition	o			
31.	LDAP Schema Description	o			
32.	Master and Shadow Access Points	o			
33.	Matching Rule Description	m			
34.	Matching Rule Use Description	m			
35.	Mail Preference	o			
36.	MHS or Address	o			Note 3
37.	Modify Rights	o			
38.	Name and Optional UID	o			Note 3
39.	Name Form Description	c3			Note 3
40.	Numeric String	o			Note 3
41.	Object Class Description	m			
42.	Octet String	o			
43.	OID	c5			Note 3
44.	Other MailBox	o			Note 3
45.	Postal Address	o			Note 3
46.	Protocol Infomration	o			
47.	Presentation Address	o			Note 3

Draft

Item No.	Protocol Element	Server		Predicate Name	Note
		Status	Support		
48.	Printable String	o			Note 3
49.	Substring Assertion	o			
50.	Subtree Specification	o			
51.	Supplier Information	o			
52.	Supplier Or Consumer	o			
53.	Supported Algorithm	o			
54.	Telephone Number	o			Note 3
55.	Teletex Terminal Identifier	o			
56.	Teletex Number	o			
57.	UTC Time	o			Note 3

c1: If [dITContent] then m else o..

c2: If [dITStruct] then m else o.

c3: If [nameForm] then m else o.

c4: If [altServer] then m else o.

c5: If [supportedExt] or [supportedControl] then m else o.

Note 1: Should use the binary encoding

Note 2: Servers must use binary transfer with this attribute syntax or if requested by the client.

Note 3: While not mandatory for LDAP servers, servers should recognize this syntax.

Note 4: Encoded using UTF-8. Only used if attribute values are carried in binary.

4.17.1 Syntax Description

Ref. RFC 2252 Para 4.3.3

Item No.	Protocol Element	Server		Predicate Name	Note
		Status	Support		
1.	numericoid	m			
2.	DESC	o			
3.	OBsolete	m			

4.17.2 RFC 1274 Syntax

Useful Syntaxes defined by the Cosine Directory Pilot.

Ref. RFC 1274 Para 9.4

Item No.	Protocol Element	Server		Predicate Name	Note
		Status	Support		
1.	caseIgnoreIA5StringSyntax	o			
2.	iA5StringSyntax	o			
3.	DNSRecordSyntax	o			
4.	NRSInformationSyntax	o			

Draft

4.18 Attributes Types

4.18.1 Operational Attribute

Ref. RFC 2252 Para 5.1

Item No.	Protocol Element	SERVER		Predicate	Note
		Status	Support		
1.	CreateTimestamp	m			Note 1
2.	ModifyTimestamp	m			Note 2
3.	CreatorsName	m			Note 1
4.	ModifiersName	m			Note 2
5.	SubschemaSubentry	m			See 4.18.1.1
6.	NamingContexts	m			Note 3
7.	AltServer	m			Note 3
8.	SupportedExtension	m		*supportedExt	Note 3,4
9.	supportedControl	m		*supportedControl	Note 3,4
10.	supportedSASLMechanisms	m			Note 3
11.	supportedLDAPVersion	m			Note 3

Note 1: Should appear in entries created using the add operation.

Note 2: Should appear in entries created using the modify operation

Note 3: Required that servers recognize attribute names but need not provide values for those attributes which correspond to features not implemented.

Note4: Predicates are only set if server implements functionality.

4.18.1.1 Subschema Entry

These attributes are typically located in the subschema entry

Ref. RFC 2252 Para 5.3

Item No.	Protocol Element	SERVER		Predicate	Note
		Status	Support		
1.	attributeTypes	m			
2.	objectClasses	m			
3.	matchingRules	m			See 4.18.1.1.1
4.	matchingRulesUse	m			See 4.18.1.1.2
5.	ldapSyntaxes	m			
6.	dITStructureRules	o		*dITStruct	See 4.18.1.2.3
7.	nameForms	o		*nameForm	See 4.18.1.2.4
8.	ditContentRules	o		*ditContent	See 4.18.1.2.5

Note 1: Server should recognize these names but only X.500 servers will implement this functionality.

4.18.1.1.1 Matching Rule Description

Ref. RFC 2252 Para 4.5

Item No.	Protocol Element	SERVER		Predicate	Note
		Status	Support		
1	numericoid	m			
2	NAME	o			
3	DESC	o			
4	OBsolete	o			
5	SYNTAX	m			

Draft

4.18.1.1.2 Matching Rule Use Description

Ref. RFC 2252 Para 4.5

Item No.	Protocol Element	SERVER		Predicate	Note
		Status	Support		
1	numericoid	m			
2	NAME	o			
3	DESC	o			
4	OBsolete	o			
5	APPLIES	m			

4.18.1.1.3 DIT Structure Rules Description

Prerequisite: [dITStruct]

Ref. RFC 2252 Para 6.33

Item No.	Protocol Element	SERVER		Predicate	Note
		Status	Support		
1	ruleIdentifier	m			
2	NAME	o			
3	DESC	o			
4	OBsolete	o			
5	FORM	m			
6	SUP	o			

4.18.1.1.4 Name Form Description

Prerequisite: [nameForm]

Ref. RFC 2252 Para 6.22

Item No.	Protocol Element	SERVER		Predicate	Note
		Status	Support		
1	numericoid	m			
2	NAME	o			
3	DESC	o			
4	OBsolete	o			
5	OC	m			
6	MUST	m			
7	MAY	o			

Draft

4.18.1.1.5 DIT Content Rule Description

Prerequisite: [ditContent]

Ref. RFC 2252 Para 6.11

Item No.	Protocol Element	SERVER		Predicate	Note
		Status	Support		
1	numericoid	m			
2	NAME	o			
3	DESC	o			
4	OBsolete	o			
5	AUX	o			
6	MUST	o			
7	MAY	o			
8	NOT	o			

4.18.2 Attribute Type Description

Ref. RFC 2252 Para 4.2

Item No.	Protocol Element	Server		Predicate Name	Note
		Status	Support		
1.	numericoid	m			
2.	NAME	o			
3.	DESC	o			
4.	OBsolete	o			
5.	SUP	o.1			
6.	EQUALITY	o			
7.	ORDERING	o			
8.	SUBSTR	o			
9.	SYNTAX	o.1		*Syntax	See 4.18.2.1
10.	SINGLE-VALUE	o			
11.	COLLECTIVE	o			
12.	NO-USER-MODIFICATION	o			
13.	USAGE	o		*Usage	See 4.18.2.2

o.1: Server should provide at least one of these fields.

4.18.2.1 Syntax Object Identifier

Prerequisite: [Syntax]

Ref. RFC 2252 Para 4.3.2

Item No.	Protocol Element	SERVER		Predicate	Note
		Status	Support		
1.	numericoid	m			
2.	len	o			

Draft

4.18.2.2 Attribute Usage

Prerequisite: [Usage]

Ref. RFC 2252 Para 4.2

Item No.	Protocol Element	SERVER		Predicate	Note
		Status	Support		
1.	userApplications	c1			
2.	directoryOperation	c1			
3.	distributedOperation	c1			
4.	dsaOperation	c1			

c1: Only one of the attribute usage elements allowed.

Draft

4.18.3 User Attribute

Ref. RFC 2256 Para 5

Item No.	Protocol Element	SERVER		Predicate	Note
		Status	Support		
1.	objectClass	m			Note 1
2.	aliasedObjectName	c1			
3.	knowledgeInformation	o			Note 2
4.	cn	o			
5.	sn	o			
6.	serialNumber	o			
7.	c	o			
8.	l	o			
9.	st	o			
10.	street	o			
11.	o	o			
12.	ou	o			
13.	title	o			
14.	description	o			
15.	searchGuide	o			Note 3
16.	businessCategory	o			
17.	postalAddress	o			
18.	postalCode	o			
19.	postOfficeBox	o			
20.	physicalDeliveryOfficeName	o			
21.	telephoneNumber	o			
22.	telexNumber	o			
23.	teletexTerminalIdentifier	o			
24.	facsimileTelephoneNumber	o			
25.	x121Address	o			
26.	internationaliSDNNNumber	o			
27.	registeredAddress	o			
28.	dstinationIndicator	o			
29.	preferredDeliveryMethod	o			
30.	presentationAddress	o			
31.	supportedApplicationContext	o			
32.	member	o			
33.	owner	o			
34.	roleOccupant	o			
35.	seeAlso	o			
36.	userPassword	o			
37.	userCertificate	o			Note 4
38.	cACertificate	o			Note 4
39.	authorityRevocationList	o			Note 4
40.	certificateRevocationList	o			Note 4
41.	crossCertificatePair	o			Note 4
42.	name	c2			Note 5
43.	givenName	o			
44.	initials	o			
45.	generationQualifier	o			
46.	x500UniqueIdentifier	o			
47.	dnQualifier	o			
48.	enhancedSearchGuide	o			
49.	protocolInformation	o			

Draft

Item No.	Protocol Element	SERVER		Predicate	Note
		Status	Support		
50.	distinguishedName	c2			Note 5
51.	uniqueMember	o			
52.	houseIdentifier	o			
53.	supportedAlgorithms	o			
54.	deltaRevocationList	o			
55.	dmdName	o			

c1: If [alias] then support is m else o.

c2: If [att_subtype] then support is o else i.

Note 1: This attribute is required by the mandatory object class, 'top'.

Note 2: Historical. No longer used.

Note 3: Historical. Made obsolete by enhancedSearchGuide.

Note 4: Stored and requested using the ;binary form.

Note 5: Normally used for defining other attributes. Normally not used to define an entry.

Draft

4.18.4 RFC 1274 Attribute

Useful Attributes defined by the Cosine Directory Pilot. Standard attributes were included by reference in this RFC, however, their definition has been superceded in RFC 2252 and RFC 2256.

Ref. RFC 1274 Para 9

Item No.	Protocol Element	SERVER		Predicate	Note
		Status	Support		
1.	userid	o			
2.	textEncodedORAddress	o			
3.	rfc822Mailbox	o			
4.	info	o			
5.	favouriteDrink	o			
6.	roomNumber	o			
7.	photo	o			
8.	userClass	o			
9.	host	o			
10.	manager	o			
11.	documentIdentifier	o			
12.	documentTitle	o			
13.	documentVersion	o			
14.	documentAuthor	o			
15.	documentLocation	o			
16.	homeTelephoneNumber	o			
17.	secretary	o			
18.	otherMailbox	o			
19.	lastModifiedTime	o			
20.	lastModifiedBy	o			
21.	domainComponent	o			
22.	aRecord	o			
23.	mXRecord	o			
24.	nSRecord	o			
25.	sOARecord	o			
26.	cNAMERecord	o			
27.	associatedDomain	o			
28.	associatedName	o			
29.	homePostalAddress	o			
30.	personalTitle	o			
31.	mobileTelephoneNumber	o			
32.	pagerTelephoneNumber	o			
33.	friendlyCountryName	o			
34.	uniqueIdentifier	o			
35.	organizationalStatus	o			
36.	janetMailbox	o			
37.	mailPreferenceOption	o			
38.	buildingName	o			
39.	dSAQuality	o			
40.	singleLevelQuality	o			
41.	subtreeMinimumQuality	o			
42.	subtreeMaximumQuality	o			
43.	personalSignature	o			
44.	dITRedirect	o			
45.	audio	o			
46.	documentPublisher	o			
47.	mhsDeliverableContentLength	o		Note 1	
48.	mhsDeliverableContentTypes	o		Note 1	
49.	mhsDeliverableEits	o		Note 1	
50.	mhsDLMembers	o		Note 1	
51.	mhsDLSumitPermissions	o		Note 1	
52.	mhsMessageStoreName	o		Note 1	

Draft

Item No.	Protocol Element	SERVER		Predicate	Note
		Status	Support		
53.	mhsORAddresses	o		Note 1	
54.	mhsPreferredDeliveryMethods	o		Note 1	
55.	mhsSupportedAutomaticActions	o		Note 1	
56.	mhsSupportedContentTypes	o		Note 1	
57.	mhsSupportedOptionalAttributes	o		Note 1	

Note 1: These attributes are defined in X.402 but included by reference in RFC 1274.

4.19 Object Classes

Ref. RFC 2252 Para 4.4 & 7

Ref. RFC 2256 Para 7

Item No.	Protocol Element	SERVER		Predicate	Note
		Status	Support		
1.	top	m			
2.	alias	c1			
3.	extensibleObject	o			
4.	subschema	m			
5.	country	o			
6.	locality	o			
7.	organization	o			
8.	organizationalUnit	o			
9.	person	o			
10.	organizationalPerson	o			
11.	organizationalRole	o			
12.	groupOfNames	o			
13.	residentialPerson	o			
14.	applicationProcess	o			
15.	applicationEntity	o			
16.	dSA	o			
17.	device	o			
18.	strongAuthenticationUser	o			
19.	certificationAuthority	o			
20.	groupOfUniqueNames	o			
21.	userSecurityInformation	o			
22.	certificationAuthority-V2	o			
23.	cRLDistributionPoint	o			
24.	dmd	o			

c1: If [alias] then support is m else o.

Draft

4.19.1 Object Classes Description

Ref. RFC 2252 Para 4.4

Item No.	Protocol Element	Server		Predicate Name	Note
		Status	Support		
1.	numericoid whsp	m			
2.	“NAME”	o			
3.	“DESC”	o			
4.	“OBSOLETE”	o			
5.	“SUP”	o			
6.	Object Class Type	o			
6.1.	ABSTRACT	c:o.1			
6.2.	STRUCTURAL	c:o.1			
6.3.	AUXILLIARY	c:o.1			
7.	“MUST”	o			
8.	“MAY”	o			

o.1: Only one type may be defined for a given Object Class. Default is STRUCTURAL.

4.19.2 PKI Object Classes

Auxiliary Object Classes for use by LDAP servers acting as PKIX repositories. May be used to manage and retrieve PKI information.

Ref 2587 Para 3.

Item No.	Protocol Element	Server		Predicate Name	Note
		Status	Support		
1.	pkiUser	o			
2.	pkiCA	o			
3.	cRLDistributionPoint	o			
4.	deltaCRL	o			

Draft

4.19.3 RFC 1274 Object Classes

Useful Object Classes defined by the Cosine Directory Pilot. Standard object classes were included by reference in this RFC, however, their definition has been superceded in RFC 2252 and RFC 2256.

Item No.	Protocol Element	Server		Predicate Name	Ref 1274 Para 8. Note
		Status	Support		
1.	pilotPerson	o			
2.	account	o			
3.	document	o			
4.	room	o			
5.	documentSeries	o			
6.	domain	o			
7.	rFC822localPart	o			
8.	dNSDomain	o			
9.	domainRelatedObject	o			
10.	friendlyCountry	o			
11.	simpleSecurityObject	o			
12.	pilotOrganization	o			
13.	pilotDSA	o			
14.	qualityLabelledData	o			
15.	mhsDistributionList	o			Note 1
16.	mhsMessageStore	o			Note 1
17.	mhsMessageTransferAgent	o			Note 1
18.	mhsOrganizationalUser	o			Note 1
19.	mhsResidentialUser	o			Note 1
20.	mhsUserAgent	o			Note 1

Note 1: These object classes are defined in X.402 but included by reference in RFC 1274.

Draft

4.19.4 Matching Rules

Ref. RFC 2252 Para 8

Item No.	Protocol Element	SERVER		Predicate	Note
		Status	Support		
1.	objectIdentifiedMatch	m			Note 1
2.	distinguishedNameMatch	o			
3.	caseIgnoreMatch	o			
4.	numericStringMatch	o			
5.	caseIgnoreListMatch	o			
6.	integerMatch	o			
7.	bitStringMatch	o			
8.	telephoneNumberMatch	o			
9.	presentationAddressMatch	o			
10.	uniqueMemberMatch	o			
11.	protocolInformationMatch	o			
12.	generalizedTimeMatch	o			
13.	caseExactIA5Match	o			
14.	caseIgnoreIA5Match	o			
15.	generalizedTimeOrderingMatch	o			
16.	caseIgnoreOrderingMatch	o			Note 2
17.	Substring Assertion	o			Note 3
17.1.	initial	co			
17.2.	any	cm			
17.3.	final	co			
18.	caseIgnoreSubstringsMatch	o			
19.	telephoneNumberSubstringsMatch	o			
20.	numericStringSubstringsMatch	o			
21.	integerFirstComponentMatch	c1			
22.	objectIdentifierFirstComponent Match	c1			

c1: If [SubschemaMod] then m else o.

Note 1: This matching rule is required by the mandatory object class, 'top'.

Note 2: Sort ordering is implementation dependent

Note 3: Used only as the syntax of assertion values in extensible match.

Draft

4.20 Rules for DN Encoding

Ref. RFC 2253

Item No.	Protocol Element	SERVER		Predicate	Note
		Status	Support		
1.	Support RDN in sequence from lower to higher separated by comma (0x2c)	m			
2.	Support multi-valued RDN separated by plus sign (0x23)	m			
3.	Support string representation of AttributeType	m			Note 1
3.1.	Support for string representation of Attributes name if from a published table	m			(RFC 2253 2.3)
3.2.	Support for dotted decimal string representation of Attribute OID if not from a published table	m			(RFC 2253 2.3)
4.	Support string representation ofAttributeValue	m			Note 1
4.1.	UTF 8 encoding for AttributeTypes defined as string	m			
4.1.1.	Escape special characters	m			Note 2
4.1.2.	Escape any character	m			Note 3
5.	Support hexadecimal BER representation ofAttributeValue	m			Note 4
6.	Support LDAP v2 encoding of DN	m			
6.1.	Allow semi-colon to be used instead of comma to separate RDNs	m			Note 5
6.2.	Allow white space before and after a RDN (i.e. on either side of comma or semi-colon).	m			Note 6
6.3.	Allow space character (0x20) to be present between RDN components	m			(RFC 2253 4)
6.4.	Allow dotted decimal string in AttributeType to be preceded with "oid"	m			Note 7
6.5.	Allow AttributeValue to be surrounded by double-quotes (0x22)	m			Note 8

Note 1: AttributeType and AttributeValue are separated by the comma character (0x3d).

Note 2: Special characters are: space (0x20) at beginning or end of string or the number sign (0x23) at beginning of string. In addition the comma, plus (0x2c), double-quote (0x22), backslash (0x5c), less than (0x3c), greater than (0x3e) and semi-colon (0x3b) character occurring anywhere in the string. The escaping mechanism is the backslash character follow one of the special characters.

Note 3: Any character in the DN may be escaped by preceding it with a backslash (0x5c) followed by the string representation of the 2 hexadecimal digits. While the RFC defines this feature as MAY servers should be able to correctly parse, so the feature is mandatory on the server side.

Note 4: Hexadecimal BER representation is preceded by the number sign (0x23). It is used if AttributeType is not defined as string or AttributeType is not in a published table (RFC 2253 2.4).

Note 5: LDAP v3 servers replace semi-colon with a comma.

Note 6: LDAP v3 servers will ignore white space between RDNs.

Note 7: "oid" is not case sensitive "OID" should also be allowed.

Note 8: Special characters will not be escaped when inside of double-quotes.

Draft

4.21 LDAP URL Definition

This document is primarily concerned with server functionality, however the following table defines support for the LDAP URL as defined in RFC 2255. The LDAP URL is a mechanism for human users to perform a search operation through a client. Normally, this capability will only be supported through browsers and not all clients need to support this capability. Clients must support and follow the URL escape mechanism as defined in RFC 2255 paragraph 3.

Ref. RFC 2255

Item No.	Protocol Element	CLIENT		Predicate	Note
		Status	Support		
1.	scheme	m			
2.	hostport	m			Note 1
3.	dn	m			Note 2
4.	attributes	m			See 4.18
5.	scope	m			Note 3
6.	filter	m			See 4.7, Note 4
7.	extensions	m			Note 5
7.1.	extension	m			Note 6
7.1.1.	extype	m			
7.1.1.1.	token	m			
7.1.1.2.	xtoken	m			Note 7
7.1.2.	exvalue	m			

Note 1: If no hostport is given, the client must have some apriori knowledge of an appropriate LDAP server to contact (RFC 2255 3)

Note 2: Identifies the base object of the LDAP search (RFC 2253 3)

Note 3: Allowable scopes are "base" for base object search, "one" for one level search, or "sub" for a subtree search. Scope of "base" is assumed if scope is omitted.

Note 4: A filter of "(objectClass=*)" is assumed when filter is omitted.

Note 5: Clients must be capable of parsing the extension mechanism from the LDAP URL, but are not necessarily required to support all extension mechanisms. Support of extensions should follow the rules defined in RFC 2255 paragraph 3.

Note 6: If the extension is supported by the client and is prefixed with a '!', then rules of criticality defined in RFC 2255 should be followed.

Note 7: xtoken uses the "x-" prefix prior to the token oid. Support for xtoken is performed by bilateral agreements between communicating parties.

4.21.1 LDAP URL Extension

Ref. RFC 2255

Item No.	Protocol Element	CLIENT		Predicate	Note
		Status	Support		
1	Bindname extension	o			
1.1	dn	c:m			Note 1

Note 1: Contains the Distinguished Name of the entry to authenticate as.